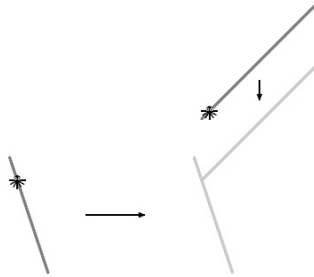


Задача А. Про любовь...

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Паук и паучиха плывут по озеру на двух веточках. Плавать они не умеют, поэтому смогут встретиться только тогда, когда веточки соприкоснутся.



Считая, что веточки имеют форму отрезков, и что они плывут с постоянными скоростями, определите, сколько осталось ждать встречи несчастным членистоногим.

Формат входных данных

Входной файл содержит 12 чисел: $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, v_{1x}, v_{1y}, v_{2x}, v_{2y}$. Координаты вершин первого отрезка: (x_1, y_1) и (x_2, y_2) , координаты вершин второго отрезка: (x_3, y_3) и (x_4, y_4) , скорость первого отрезка (v_{1x}, v_{1y}) , скорость второго отрезка (v_{2x}, v_{2y}) . Все числа целые и не превосходят по модулю 10^4 . В начальный момент времени веточки не соприкасаются.

Гарантируется, что веточки имеют ненулевую длину.

Формат выходных данных

Выведите в выходной файл время до ближайшего момента, когда веточки соприкоснутся, с ошибкой не более 10^{-4} . Если веточки не соприкоснутся никогда, выведите число -1 .

Примеры

стандартный ввод	стандартный вывод
0 0 -1 3 4 4 7 7 3 0 0 -1	1.6000000000
0 0 -1 3 4 4 7 7 1 0 0 -3	-1

Задача В. Теодор Рузвельт

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

«Теодор Рузвельт» — флагман военно-морского флота Кукуляндии. Заклятые враги кукуляндцев, флатландцы, решили уничтожить его. Они узнали, что «Теодор Рузвельт» представляет собой выпуклый многоугольник из n вершин и узнали его координаты. Затем они выпустили m баллистических ракет и определили координаты точек, где эти ракеты взорвались. По расчётам штаба флатландцев, «Теодор Рузвельт» будет уничтожен, если в него попадёт хотя бы k ракет. Вычислите, удалось ли флатландцам уничтожить корабль.

Формат входных данных

В первой строке через пробел записаны целые числа n, m, k ($3 \leq n \leq 10^5, 0 \leq k \leq m \leq 10^5$). В последующих n строках записаны координаты вершин многоугольника в порядке обхода против часовой стрелки. В следующих m строках записаны координаты точек. Гарантируется, что все координаты — целые числа, не превосходящие по модулю 10^9 .

Формат выходных данных

Выведите «YES», если в многоугольнике или на его границе лежит по крайней мере k точек, и «NO» в противном случае.

Пример

стандартный ввод	стандартный вывод
5 4 2 1 -1 1 2 0 4 -1 2 -1 -1 -2 -1 1 -1 0 1 2 3	YES

Задача С. Парк

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В столице Байтленда есть огороженный парк, территория которого представляет собой прямоугольник. Деревья и посетители в парке представлены в виде кругов.

В парке четыре входа, по одному в каждом углу (1 — внизу слева, 2 — внизу справа, 3 — вверху справа, 4 — вверху слева). Посетители могут входить и выходить из парка только через входы. Посетители могут входить и выходить из парка, когда они касаются обеих сторон угла соответствующего входа. Посетители могут свободно перемещаться по парку, но они не могут перекрывать деревья и заборы.

Ваша задача — рассчитать для каждого посетителя, учитывая вход, в который они войдут в парк, через какие выходы они могут выйти из парка.

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($1 \leq n \leq 2000$ $1 \leq m \leq 100\,000$) — количество деревьев в парке и количество посетителей.

Вторая строка ввода содержит два целых числа w и h ($1 \leq w, h \leq 10^9$) — ширину и высоту парковой зоны. Нижний левый угол имеет координаты $(0, 0)$, а верхний правый угол имеет координаты (w, h) .

После этого есть n строк, которые описывают деревья. Каждая строка содержит три целых числа x , y и r , что означает, что центр i -го дерева (x, y) и его радиус r . Деревья не перекрывают друг друга или забор.

Наконец, есть m строк, которые описывают посетителей. Каждая строка содержит два целых числа r и e — радиус посетителя и вход, в который они войдут в парк.

Кроме того, ни одно дерево не перекрывает квадратную область $2k \times 2k$ в каждом углу, где k — радиус наибольшего посетителя.

Формат выходных данных

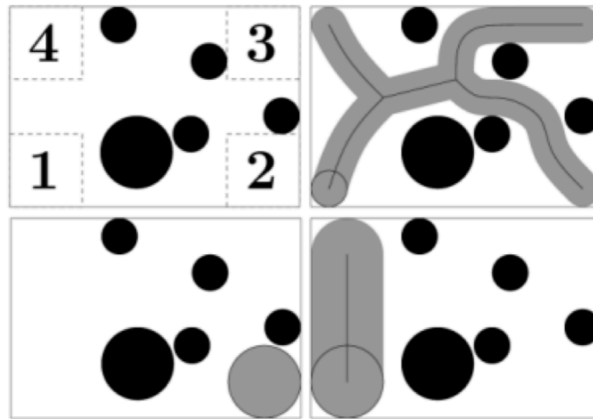
Вы должны вывести для каждого посетителя одну строку, содержащую входы, через которые они могут выйти из парка, в отсортированном порядке без пробелов между ними.

Пример

стандартный ввод	стандартный вывод
5 3	1234
16 11	2
11 8 1	14
6 10 1	
7 3 2	
10 4 1	
15 5 1	
1 1	
2 2	
2 1	

Замечание

На следующем рисунке показаны зоны входа и возможные маршруты для каждого посетителя:



Задача D. Большой треугольник

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 8 секунд
Ограничение по памяти: 1024 мегабайта

Есть странная особенность: если соединить между собой города Ростов, Таганрог, Шахты, то получится треугольник

«Невероятно, но факт»

В ЛКШ приезжают ученики из самых разных уголков России и зарубежья. Вы отметили на карте все города, откуда приехали участники ЛКШ.

Затем, вы решили подготовить интересную инфографику на основе этой карты. Первое что вы захотели сделать — это найти три города на этой карте, так чтобы получился треугольник площади ровно S .

Формат входных данных

В первой строке входных данных находится два целых числа n и S ($3 \leq n \leq 3333$, $1 \leq S \leq 2 \cdot 10^{18}$) — количество городов на карте и требуемая площадь треугольника.

В следующую n строках находится описание городов, по одной на строке. Каждый город описывается своими координатами x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$).

Гарантируется, что все города находятся в различных точках. Также гарантируется, что никакие три города не лежат на одной прямой.

Формат выходных данных

Если решения не существует — выведите «No» (без кавычек).

Иначе выведите «Yes» (без кавычек), после чего выведите три пары координат — координаты городов, образующих треугольник площади S .

Примеры

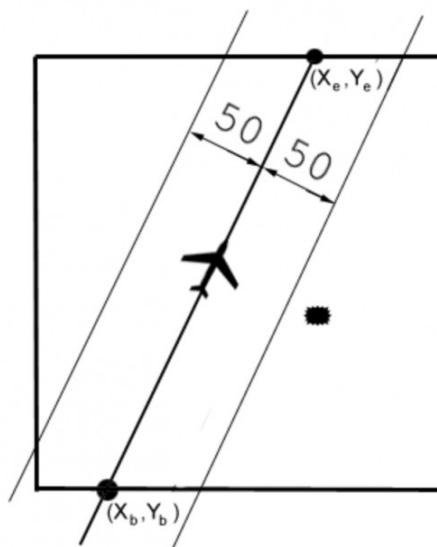
стандартный ввод	стандартный вывод
3 7 0 0 3 0 0 4	No
4 3 0 0 2 0 1 2 1 3	Yes 0 0 1 3 2 0

Задача Е. Лесной пожар

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

В МЧС поступило сообщение о возможном лесном пожаре в заданном квадрате тайги. Для поиска места возгорания было послано N самолетов. Однако ни один из экипажей пожар не обнаружил.

Известно, что с самолета видна полоса тайги, границы которой находятся на расстоянии 50 км справа и слева от той линии на поверхности Земли, над которой пролетает самолет (см. рисунок), причем точки, находящиеся на расстоянии ровно 50 км от этой линии, все еще видны. Донесение с каждого самолета содержало информацию о том, в каких двух различных точках (x_b, y_b) и (x_e, y_e) самолет входил в заданный квадрат и покидал его соответственно. Между этими точками самолет двигался строго по прямой.



Требуется написать программу, которая определит, действительно ли весь заданный квадрат тайги был просмотрен с самолетов. Если это не так, то программа должна находить координаты какой-нибудь точки, находящейся внутри или на границе квадрата и не попавшей ни в одну из просмотренных полос.

Формат входных данных

В первой строке записано натуральное число L — размер заданного квадрата тайги в километрах ($0 \leq L \leq 1000$).

Во второй строке — натуральное число N ($1 \leq N \leq 100$) — количество самолетов.

В каждой из последующих N строк записано донесение с самолета — четыре вещественных координаты x_b, y_b, x_e, y_e . Координаты заданы в километрах. Стороны квадрата тайги параллельны осям координат, его левый нижний угол находится в точке с координатами $(0, 0)$, а правый верхний — в точке (L, L) .

Формат выходных данных

Если заданный квадрат был просмотрен полностью, то выведите единственное слово «YES», написанного заглавными латинскими буквами. В противном случае выведите слово «NO», а в следующей строке выведите через пробел координаты x и y какой-либо точки, которая не попала ни в одну из просмотренных полос. Координаты нужно выводить в километрах с ошибкой не более одного метра (в одном километре 1000 метров).

Примеры

стандартный ввод	стандартный вывод
245 1 26.1 0 193.568 245	NO 155.123 100
50 1 0 25 25 25	YES

Замечание

Вы можете написать тут нормальное решение и умереть. А можете вспомнить про точку в квадрате, непокрытую окружностями.

Задача F. Бульдозер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

Королевство JOI известно добычей золота. Каждый год они используют бульдозер для его добычи.

Земля королевства описывается плоскостью с декартовыми координатами. Есть n мест, i -е место имеет координаты (x_i, y_i) . В каждом месте возможно добыть золото или камни, но нельзя одновременно и то, и другое.

Если в i -м месте можно добыть золото, то его ценность там равна v_i . Если же там камни, то, чтобы избавиться от них, необходимо потратить c_i .

Мы используем бульдозер следующим образом. Сначала мы выбираем две параллельные прямые, затем добываем всё золото и камни, находящиеся между этими прямыми, включая их границы.

Выгода равна сумме ценностей золота за вычетом трат на избавление от камней. Необходимо максимизировать выгоду.

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 2000$) — количество мест.

Далее следуют n строк, i -я из которых содержит три целых числа x_i, y_i, w_i ($|x_i|, |y_i|, |w_i| \leq 10^9$, $w_i \neq 0$) — координаты места (x_i, y_i) и специальное число.

- Если $w_i \geq 1$, то i -е место содержит золото, и $v_i = w_i$.
- Если $w_i \leq -1$, то i -е место содержит камни, и $c_i = -w_i$.

Гарантируется, что все точки различны.

Формат выходных данных

Выведите одно число — ответ.

Примеры

стандартный ввод	стандартный вывод
5 -5 5 -2 2 5 10 1 4 -2 4 -5 4 -2 2 7	19
6 0 0 6 1 0 -2 2 0 8 0 1 -2 1 1 5 2 1 -2	15
5 0 0 2 4 0 2 3 2 -1 1 2 2 1 1 -1	5
2 0 0 -1 1 0 -1	0
15 10 3 30 5 10 -17 4 -5 14 0 -3 -9 -2 3 17 6 9 -19 -9 -6 -14 -2 -3 10 -3 -3 30 8 1 -28 9 -9 -5 7 -5 -24 -8 -10 5 -7 2 20 10 -3 -13	107